**Problem Description**

Create a menu-driven that has the following options:

* Create a vector that contains a certain amount of random numbers, where the user defines the amount of numbers and the range that these numbers fall between.
* Print the vector
* Clear the vector
* Find the mean value in the vector
* Find the median value in the vector
* Find the modal values in the vector.

**Input & Output**

|  |  |
| --- | --- |
| **Input** |  |
| *Input Description* | *Mechanism* |
| Choice | Standard Input Stream |
| Length of vector | Standard Input Stream |
| Minimum value | Standard Input Stream |
| Maximum value | Standard Input Stream |

|  |  |
| --- | --- |
| **Output** |  |
| *Output Description* | *Stream(Optional)* |
| Menu(Options a,b,c,d,e,f and x) | Standard Output Stream |
| Conversion error for integer | Standard Error Stream |
| Vector | Standard Output Stream |
| Mean | Standard Output Stream |
| Median | Standard Output Stream |
| Modal values | Standard Output Stream |

**Data Format**

|  |  |  |
| --- | --- | --- |
| *Identifier* | *Product Type* | *Description* |
| blnContinue | Bool | Used to exit menu |
| chChoice | Char | Users choice from menu |
| vecNums | vector | Vector of random numbers |
| intCount | integer | Length of vector |
| intUpper | integer | Upper limit of range |
| intLower | integer | Lower limit of range |
| dblMean | double | Mean value |
| intNum | integer | Random number |

**Pseudo Code**

Include “libprac6.h”

Using namespace std

Using namespace VectorSpace

srand()

blnContinue -> true

vecNums

do

system clear

Output Menu

chChoice -> Input

switch(chChoice)

case ‘a’

intCount -> Input

intUpper -> Input

intLower -> Input

vecNums -> GenVec

break

case ‘b’

PrintVector

Break

case ‘c’

vecNums.clear

break

case ‘d’

GetMean

*void GetMean(…)*

*{*

*dblMean -> 0.0*

*intTotal -> 0*

*intLength -> 0*

*for(int n: vecNums)*

*{*

*intTotal +-> n*

*intLength +-> 1*

*}*

*dblMean -> intTotal / intLength*

*Output dblMean*

*}*

break

case ‘e’

GetMedian

break

case ‘f’

GetMod

Break

case ‘x’

blnContinue -> false

break

default

error

system(“clear”)

while(blnContinue)

**UML Activity Diagram**

